

Simplifying Wireless Social Caching

Mohammed Karmoose, Martina Cardone, and Christina Fragouli

Abstract—Social groups give the opportunity for a new form of caching. In this paper, we investigate how a social group of users can jointly optimize bandwidth usage, by each caching parts of the data demand, and then opportunistically share these parts among themselves upon meeting. We formulate this problem as a Linear Program (LP) with exponential complexity. Based on the optimal solution, we propose a simple heuristic inspired by the bipartite set-cover problem that operates in polynomial time. Furthermore, we prove a worst case gap between the heuristic and the LP solutions. Finally, we assess the performance of our algorithm using real-world mobility traces from the MIT Reality Mining project dataset and two mobility traces that were synthesized using the SWIM model. Our heuristic performs closely to the optimal in most cases, showing a better performance with respect to alternative solutions.

Index Terms—Social networks, wireless networks, cooperative caching, linear programming, set-cover problem, polynomial-time heuristic.

1 INTRODUCTION

TODAY, a considerable fraction of data requirements in wireless networks comes from “social groups”. Members of a social group share common interests/goals and exhibit frequent and regular meeting patterns. Situations may arise where accommodating the data requirements of a social group through the wireless network is highly costly and infeasible. Examples of such scenarios are: (1) a group of students attending an online-course in an economically-challenged country, where it is costly to download the material that each student needs; (2) a group of tourists interested in obtaining touristic advertising and videos in a foreign country, where it is expensive to have cellular data connection; (3) in the aftermath of catastrophic emergencies, where the infrastructured networks are compromised and it is infeasible to establish stable connections with citizens. These examples highlight the critical importance of reducing the dependence on infrastructured networks. By exploiting social interactions among group members, it becomes possible to distribute the downloading efforts among the members who can then exchange data through local and cost-free connections.

We consider a social group of N members who all wish to acquire (within a time period of duration t) a set of M files on their smart wireless devices. These M files are stored on a server to which the N users have access through a wireless communication link. Examples of this type of scenarios include co-workers downloading files needed before a meeting, conference participants downloading presentations for next sessions, students downloading class materials and sport fans downloading videos during an event. We assume that the group members have regular meeting patterns, which are correlated with the group activity (e.g., work, sport, entertainment); we model these meeting patterns as random events. In particular, we assume that with some probability, members meet each other (one or multiple times) within the period of interest.

In this work we seek to minimize the usage of the bandwidth. As supported by almost all smart devices today, we assume that users can connect either directly to the server through a longhaul connection (e.g., cellular), which is expensive in bandwidth, or to each other, when in physical proximity, through a local and cost-free Device-to-Device (D2D) connection (e.g., Bluetooth). At the beginning of the period, each member downloads a certain amount of the files through the longhaul (bandwidth expensive) connection and locally caches this information. When two (or more) users meet, they exchange what they have in their caches using local (cost-free) connections. We consider two variations: in the *direct* case, users share only the data they themselves have downloaded (e.g., because of liability/authentication reasons), while in the *indirect* case, users share both the data they themselves have downloaded as well as the data they have collected through previous encounters with other members. At the end of the time period of duration t , if a user has not received yet all the files, she will download the missing amount of data through the longhaul connection. The fundamental question we seek to answer is the following: at the beginning of the period, how much should each user download through the longhaul connection, so that the expected total usage of bandwidth within the period is minimized?

Related Work. Distributed and cooperative caching, as a means of improving the system performance, has received considerable attention lately as summarized next.

Work in the literature has considered the ultimate information-theoretic performance [1], [2], [3]. The common objective of these works is to find the optimal caching policy in a scenario where different users have different demands, where the demands may be uniform [1] or not [2], [3]. In all these works the amount of caching is known and the randomness lies in the users demands, while in our scenario the randomness lies in the member encounters.

In a situation where a group of smartphone users, with a common and simultaneous demand, are within proximity, cooperative caching is closely related to cooperative downloading [4], [5], [6]. The key-ingredient of these works,

The authors are with the Electrical Engineering Department, University of California, Los Angeles (UCLA), CA 90095 USA (e-mail: mkarmoose@ucla.edu, martina.cardone@ucla.edu, christina.fragouli@ucla.edu). M. Karmoose was supported by NSF under Award #1423271. M. Cardone was supported by NSF under Award #1321120.

similar to ours, is that each user downloads parts of the content from the server (through a longhaul connection) and then disseminates (through a Wi-Fi connection) these parts to users in proximity. Distinct from these works, we do not a priori assume that users within the same group will meet and be able to exchange data within the prescribed period.

In a scenario where cooperative caching is allowed, a natural question arises on how to create proper incentives for the different users to cache previously downloaded content, which potentially is not any more useful. This problem has been analyzed, e.g., in [7], [8], [9]. In our framework, since users have a common demand, there is no rebate cost on communication within a group and members are always enticed to cache content, leading to distinct algorithms.

Cooperative caching has also been analyzed in the context of delay tolerant networks. In [10], [11], the authors derive the optimal caching policy that maximizes the *social welfare*, i.e., the average system utility. This metric is a function of other factors, e.g., users impatience and the popularity of the files. In [12], the authors aim to minimize the average delay and/or the average consumed energy. This is achieved by letting the server send random linear combinations of data packets to the users, and then - through heuristic algorithms - determine a set of *qualified* users to broadcast the transmissions to others. The differentiating feature of our work, however, lies in the objective function we optimize for: the number of downloads from the server. This implies that in our scenario, even if the members always have access to the longhaul link, they would anyway wait until the end of the time period before downloading from the server. In contrast, the incentive in [10], [11] would cause the users to download from the server whenever they have access, while the objective in [12] is to minimize the average consumed energy and the average delay.

Our work is similar to data offloading in cellular delay-tolerant networks: here, the goal is to reduce cellular data traffic by opportunistically sharing data among end-users through Terminal-To-Terminal (T2T) communications (we refer to [13] for a comprehensive study on this topic). A widely used approach is the so-called “subset selection”, where the central coordinator (i.e., the server) selects a subset of users to route the required data to other users in the network. In [14], the authors propose a target-set approach, where the server selects k users, with the goal to maximize the number of reached users (through T2T connections). Since this problem is NP-hard, the authors propose a sub-optimal greedy heuristic. The authors in [15] study the regular interaction patterns among users to predict the VIP users (i.e., those who experience the highest number of meetings); these are then selected to be the local data forwarders. Distinct from these works: (i) we show that, by allowing users to cache network-coded parts of the data, the problem can be formulated as an easy-to-handle Linear Program (LP); (ii) thanks to the rigorous mathematical formulation, we prove an analytical performance guarantee of the proposed caching strategies; (iii) by means of numerical evaluations on real data, we present scenarios in which our approach achieves a better performance with respect to [14].

Contributions. We first formulate our problem as an LP, which allocates amounts of data to download to each mem-

ber so as to minimize the expected total cost (total number of downloads). Towards this goal, we assume that the data is coded (as in network coding [16]). Since each user caches randomly coded data segments, it is unlikely that two different caches have the same content. Thus, a user receives novel information whenever she accesses a cache to which she has not had access before. With this, for N members, we have $2^{\binom{N}{2}}$ possible meeting patterns, each occurring with a certain probability. The LP is hence of exponential size. We perform several simplification steps and prove that, in the symmetric case, i.e., when all pairs of members meet with equal probability, the complexity of the solution reduces to linear in N . Moreover, through an artifact, we show how the indirect case can be studied within the framework of the direct case without the need to develop a separate one.

We then show a surprising connection between our problem and the well-known *set-cover* problem. In particular, we prove that the solution of the optimal LP is lower bounded by the weighted sum of the solutions of several set-cover problems. Each problem is described by an adjacency matrix, which is related to a possible meeting pattern among the users; the weight depends on the probability that this particular meeting pattern occurs.

Next, inspired by the structure of the solution of the optimal LP, we propose a simple polynomial-time approximation algorithm that we name AlgCov. AlgCov is related to the bipartite set-cover problem, reduces to a closed form expression in the symmetric case, and achieves in our simulations a performance close to the optimal. Moreover, by using approximation techniques and tools from LP duality, we analytically prove that AlgCov outputs a solution that is at most an additive worst-case gap apart from the optimal; the gap depends on the number of members and on the probability that the users meet.

Finally, we evaluate the performance of AlgCov over real-world datasets. We use data from the MIT Reality Mining project [17], as well as two synthesized mobility traces, generated by the SWIM model [18]: a simulation tool used to synthesize mobility traces of users based on their social interactions. These synthesized traces were created based on real mobility experiments conducted in IEEE INFOCOM 2005 [19] and Cambridge in 2006 [20]. We assess the performance over the case where group members exhibit relatively symmetric meeting patterns (i.e., users have approximately the same expected number of users to meet) as well as asymmetric patterns (i.e., different users have different expected number of users to meet). For both configurations, AlgCov achieves a performance close to the optimal. AlgCov performance is also compared with alternative solutions, e.g., the target-set heuristic in [14] and CopCash, a strategy which incorporates the concept of caching into the cooperative downloading approach proposed in [5]. This paper is based on the work in [21], with the following novel contributions: (i) proofs of the theorems in [21], (ii) Theorems 3.1, 3.3 and 6.1, (iii) connection to the set-cover problem, (iv) CopCash comparison, and (v) SWIM model experiments.

Paper Organization. Section 2 introduces our problem. Section 3 formulates the problem as an exponentially complex LP and shows that this complexity becomes linear in N in the symmetric case. Section 4 shows the connection of the

LP formulation to the set-cover problem. Section 5 proposes two polynomial time heuristics, based on which we design AlgCov in Section 6. Section 6 also derives an additive gap bound on AlgCov from the optimal solution. Section 7 evaluates the performance of AlgCov over real-world and synthesized traces; Section 7 also provides comparisons with alternative solutions. Finally Section 8 concludes the paper. Some of the proofs can be found in the Appendix.

Notation. Lower and upper case letters indicate scalars, boldface lower case letters denote vectors and boldface upper case letters indicate matrices; calligraphic letters indicate sets; $|\mathcal{A}|$ is the cardinality of \mathcal{A} , $\mathcal{A} \setminus \mathcal{B}$ is the set of elements that belong to \mathcal{A} but not to \mathcal{B} and $\text{Pow}(\mathcal{A})$ is the power set of \mathcal{A} ; $[n_1 : n_2]$ is the set of integers from n_1 to $n_2 \geq n_1$; $[x]^+ := \max\{0, x\}$ for $x \in \mathbb{R}$; $\mathbb{E}[\cdot]$ is the expected value; $\mathbf{1}_j$ (respectively, $\mathbf{0}_j$), is a j -long column vector of all ones (respectively, zeros); \mathbf{A}^T is the transpose of the matrix \mathbf{A} ; $\mathbb{1}_{\{P\}}$ is the indicator function, i.e., it is equal to 1 when statement P is true and 0 otherwise.

2 SETUP

Goal. We consider a set \mathcal{N} of N users $u_i \in \mathcal{N} \forall i \in [1 : N]$ who form a social group. All users need to obtain the same set \mathcal{M} of $|\mathcal{M}| = M$ information units (files), that are available from a server, within the same time period of duration t . Users can access the server through a direct longhaul wireless link that has a cost c per downloaded information unit. They can also exchange data with each other through a cost free D2D communication link, when (and if) they happen to physically encounter each other - when their devices can directly connect to each other, e.g. through Bluetooth. Our goal is to minimize the average total downloading cost across the user group. Clearly, with no cooperation, the total cost is NMc .

Assumptions. We make the following assumptions.

- Complete encounter cache exchange. According to [22], the average contact duration between two mobile devices is 250 seconds, sufficient for delivering approximately 750 MBs using standard Bluetooth 3.0 technology. Thus, we assume that encounters last long enough to allow the users who meet to exchange their whole cache contents.
- No memory constraints. Since the users demand the whole \mathcal{M} , we assume they have sufficient cost-free storage for it.
- A-priori known Bernoulli distribution. We assume that the pairwise meetings between the users (i) are Bernoulli distributed and (ii) occur with probabilities that are known a priori. Studies in the literature have been conducted to provide mobility models for users based on their social interactions (see [23] and references therein). While such models are fit for simulation purposes, they appear complex to study from an analytical point of view. Thus, we make assumption (i) as a means to derive closed-form solutions and provide analytical performance guarantees; we also assess the validity of our derived solutions on synthesized mobility traces which use mobility models. Assumption (ii) can be attained by exploiting the high regularity in human mobility [24], [25] to infer future meeting probabilities based on previous meeting occurrences.
- Delay-tolerant users. Even with a longhaul connection,

users can endure a delay, at most of duration t , in data delivery so as to receive data via D2D communications.

- Network coded downloads. We assume that users download linear combinations of the information units [16].

Approach. Our scheme consists of three phases, namely

- 1) *Caching phase*: before the period of duration t starts, each user downloads a (possibly different) amount x_i of the file set \mathcal{M} using the longhaul connection at cost cx_i . In our LP formulations, we assume, without loss of generality, that $|\mathcal{M}| = M = 1$, and thus x_i is a fraction.
- 2) *Sharing phase*: when two or more users meet, they opportunistically exchange the data they have in their caches. We consider two separate cases: the *direct* sharing case, where users share data they themselves have downloaded from the server (e.g., because of liability/authentication reasons), and the *indirect* sharing case, where users also exchange data they have collected through previous encounters.
- 3) *Post-sharing phase*: each user downloads the amount y_i she may be missing from the server at a cost cy_i . In the LPs, since we assume $M = 1$, we have that $0 \leq y_i \leq 1$.

With this approach, what remains is to find the optimal caching strategy. For instance, it is not obvious whether a user, who we expect to meet many others, should download most of the file (so that she delivers this to them) or almost none (as she will receive it from them). Moreover, downloading too much may lead to unnecessary cost in the caching phase; downloading too little may lead to not enough cost-free sharing opportunities, and thus unnecessary cost in the post-sharing phase.

3 LP FORMULATIONS

We formulate an LP that takes as input the encounter probabilities of the users, and finds $x_i, \forall i \in [1 : N]$ that minimize the average total cost during the caching and post-sharing phases. We consider direct and indirect sharing.

Direct Sharing. During encounters users can exchange what they have personally downloaded from the server. Thus, whether users u_i and u_j meet each other multiple times or just once during the period of duration t , they can still only exchange the same data - multiple encounters do not help.

We model the encounters between the N users as a random bipartite graph $(\mathcal{U}, \mathcal{V}, \mathcal{E})$, where: (i) \mathcal{U} contains a node for each of the N users at the caching-phase, (ii) \mathcal{V} contains a node for each of the N users at the end of the period of duration t , and (iii) an edge $e \in \mathcal{E}$ always exists between a node and itself and it exists between (u_i, u_j) , with $i \neq j$, with probability $p_{i,j}^{(t)}$; this edge captures if u_i and u_j meet each other (one or multiple times) during the period of duration t and share their cache contents. There are $K = 2^{\binom{N}{2}}$ realizations (configurations) of such a random graph, indexed as $k = [1 : K]$. Each configuration has an adjacency matrix $\mathbf{A}^{(k)}$ and occurs with probability $p_k^{(t)}$. For brevity, in what follows we drop the superscript (t) .¹

1. With this formulation, we can directly calculate the probabilities $p_k, \forall k \in [1 : K]$, if the pairwise encounters are independent and Bernoulli distributed with probabilities $p_{i,j}$; however, the Bernoulli assumption is not necessary, since the formulation only uses the probabilities $p_k, \forall k \in [1 : K]$, that could be provided in different ways as well. We also remark that $p_k, \forall k \in [1 : K]$, not only depends on the duration t , but also on the start of the sharing period.

We denote with $\mathbf{x} = [x_{[1:N]}]^T$ the vector of the downloaded fractions and with $\mathbf{r}^{(k)} = [r_{[1:N]}^{(k)}]^T$ the vector of the received fractions after the sharing phase for the k -th configuration. With this we have $\mathbf{r}^{(k)} = \mathbf{A}^{(k)}\mathbf{x}$. The cost of post-sharing downloading in the k -th configuration is $\mathbf{C}^{(k)} = [\mathbf{C}^{(k)}(1), \dots, \mathbf{C}^{(k)}(N)]^T$ with

$$\mathbf{C}^{(k)}(i) = c \cdot \max\{0, 1 - r_i\} \forall i \in [1 : N].$$

With the goal to minimize the total cost (i.e., caching and post-sharing phases) incurred by all users, the optimal \mathbf{x} becomes the solution of the following optimization problem

$$\begin{aligned} \min_{\mathbf{x}} \quad & \sum_{k=1}^K p_k \sum_{i=1}^N \mathbf{C}^{(k)}(i) + c \cdot \mathbf{1}_N^T \mathbf{x} \\ \text{subject to} \quad & \mathbf{x} \geq \mathbf{0}_N, \end{aligned}$$

or equivalently

$$\begin{aligned} \min_{\mathbf{x}, \mathbf{y}} \quad & f^{\text{Opt}}(\mathbf{x}, \mathbf{y}) = \sum_{k=1}^K p_k \sum_{i=1}^N y_{i,k} + \mathbf{1}_N^T \mathbf{x} \\ \text{subject to} \quad & \mathbf{x} \geq \mathbf{0}_N, \mathbf{y} \geq \mathbf{0}_{N \times K}, \\ & \mathbf{1}_N - \mathbf{A}^{(k)} \mathbf{x} \leq \mathbf{y}_k, \forall k \in [1 : K], \end{aligned} \quad (1)$$

where the variable $y_{i,k}$ represents the fraction to be downloaded in the post-sharing phase by user $i \in [1 : N]$ in configuration $k \in [1 : K]$ after receiving data from the users encountered in the sharing phase. Without loss of generality, we assumed $c = 1$. The LP formulation in (1) has complexity $\mathcal{O}(2^{N^2})$ (due to the $K = 2^{\binom{N}{2}}$ possible realizations over which we have to optimize), which prohibits practical utilization - yet this formulation still serves to build intuitions and offers a yardstick for performance comparisons.

The following theorem provides an alternative formulation of the LP in (1), which reduces the complexity to $\mathcal{O}(2^N)$. Let π_v be any row vector of length N with zeros and ones as entries. By excluding the all-zero vector, there are $2^N - 1$ such vectors, which we refer to as *selection vectors*. We let \mathcal{T} be the set of all such vectors and \mathcal{S}_v be the set of users corresponding to the selection vector π_v .

Theorem 3.1. *Let $\pi_v \in \mathcal{T}$, $\forall v = [1 : 2^N - 1]$. Then the LP in (1) can be equivalently formulated as*

$$\begin{aligned} \min_{\mathbf{x}, \mathbf{y}} \quad & \sum_{v=1}^{2^N-1} \left(\sum_{u \in \mathcal{S}_v} \Pr(u \rightarrow \mathcal{S}_v) \right) y_v + \mathbf{1}_N^T \mathbf{x} \\ \text{subject to} \quad & \mathbf{x} \geq \mathbf{0}_N, \mathbf{y} \geq \mathbf{0}_{N \times K}, \\ & 1 - \pi_v \mathbf{x} \leq y_v, \forall v \in [1 : 2^N - 1], \end{aligned} \quad (2)$$

where $\Pr(u \rightarrow \mathcal{S}_v)$ is the probability that user u is connected to all and only the users in \mathcal{S}_v .

The main observation behind the proof of Theorem 3.1 (see Appendix A) is that the LP in (1) has an inherent symmetry: the Left-Hand-Sides (LHS) of the constraints of the LP in (1) are all repetitions of constraints of the form $1 - \pi_v \mathbf{x}$. Thus, an optimal solution will let the right-hand-side of the constraints with the same LHS be equal. By appropriately grouping these constraints and variables together, we arrive at the LP in (2) which has complexity of $\mathcal{O}(2^N)$.

LP for the symmetric case

We now assume that users meet pairwise with the same probability, i.e., $p_{i,j} = p$, $\forall (i, j) \in [1 : N]^2, i \neq j$ during the period of duration t . Thus, p_k only depends on the number of encounters (as opposed to which exactly) that the configuration k contains. Many realistic scenarios can be modeled as symmetric (e.g., students in the same class, doctors in the same medical department in a hospital, military soldiers in the battlefield). The next theorem (whose proof is provided in Appendix B) significantly simplifies the problem in (1).

Theorem 3.2. *In the symmetric scenario, the LP in (2) can be simplified to the following LP*

$$\begin{aligned} \min_{x, y_i, i \in [1:N]} \quad & \sum_{i=1}^N y_i \binom{N-1}{i-1} N p^{i-1} (1-p)^{N-i} + Nx \\ \text{subject to} \quad & x \geq 0, y_i \geq 0, \forall i \in [1 : N], \\ & 1 - ix \leq y_i, \forall i \in [1 : N]. \end{aligned} \quad (3)$$

The LP in (3) has linear complexity in N , i.e., the optimal solution is obtained in polynomial time. It is worth noting that the symmetric assumption is made to get an analytical handle on the problem. When we assess the performance on real datasets we will relax this assumption by requiring users to have an approximately equal average degree (i.e., number of encountered users), as explained in Section 7.

Indirect Sharing. Enabling users to share both what they downloaded from the server as well as what they received from previous encounters, gives rise to interesting situations, since now, not only multiple encounters help, but also the order of the encounters matters. Assume for instance that, during the period of duration t , u_1 meets u_2 , and later u_2 meets u_3 . Now u_3 will have ‘indirectly’ received x_1 as well as x_2 . If instead, u_2 meets u_3 before she meets u_1 , then u_3 will only receive x_2 , but u_1 will receive both x_2 and x_3 . Moreover, if u_2 again meets u_3 later during the period, u_3 can receive x_1 through this second encounter with u_2 .

To model sequential encounters, we split the time period of duration t into T time segments, such that, during each segment, it is unlikely for more than one encounter opportunity to occur (note that one user can still meet multiple people simultaneously). We then ‘expand’ over time our bipartite graph to a $(T+1)$ -partite layered graph, by adding one layer for each time segment, where the ℓ -th time segment corresponds to the duration between times $t_{\ell-1}$ and t_ℓ , with $\ell \in [1 : T]$. In contrast to the direct case, at the end of the period of duration t , node u_j is able to receive x_i from node u_i , if and only if there exists a path connecting u_i at the first layer to u_j at the last layer; u_i and u_j do not need to have directly met, provided that such a path exists.

Note that in the bipartite (direct) case, the probability $p_{i,j}$ (respectively $p_{j,i}$) associated with the edge from user i to user j (respectively, from j to i) indicates how often user i shares her cache content with user j (respectively, j with i), with $p_{i,j} = p_{j,i}$. Thus, using this time-expanded model, the indirect case can be readily transformed into an equivalent bipartite (direct) case, by replacing the probability of each two users meeting in the bipartite graph, with the probability of a path existing between these two users on the $(T+1)$ -partite graph. Let t_0 and t_T be the time instants at which the $(T+1)$ -partite graph begins and ends, respectively. Let

$P_N^{(T)}(u \rightarrow S_v; t_0)$ be the probability that, in the time interval between t_0 and t_T , a path exists between user u and each of the users inside the set S_v . We let $p_{i,j}^{(t_\ell)}, \forall \ell = [0 : T-1]$ be the probability that users i and j are connected between time instants t_ℓ and $t_{\ell+1}$. Given this, the next theorem derives the values of $P_N^{(T)}(u \rightarrow S_v; t_0)^2$.

Theorem 3.3. Assume a $(T+1)$ -partite model, where $t_{\ell-1}$ and t_ℓ are respectively the starting and ending times of the ℓ -th time segment, $\forall \ell \in [1 : T]$. Let \mathcal{N} be the set of all users, and let $S_I \subseteq \mathcal{N}$ and $S_O \subseteq \mathcal{N}$ be two sets of users of sizes I and O , respectively. Let $\mathcal{U} = S_O \setminus S_I$. Denote with $S_I \rightarrow S_O$ the event of having the users in S_I meeting exactly the users in S_O and let $P_N^{(n+1)}(S_I \rightarrow S_O; t_\ell)$ be the probability of this event happening between time instants t_ℓ and $t_{\ell+n+1}$. Then, for $(\ell, n) \in [0 : T-1]^2, \ell + n \leq T-1$, this probability is given by

$$P_N^{(n+1)}(S_I \rightarrow S_O; t_\ell) = \sum_{u \in \text{Pow}(\mathcal{U})} P_N^{(1)}(S_I \rightarrow S_I + u; t_{\ell+n}) \cdot P_N^{(n)}(S_I + u \rightarrow S_O; t_\ell),$$

where

$$P_N^{(1)}(S_I \rightarrow S_O; t_n) = \prod_{a \in \mathcal{N} \setminus S_O} \prod_{b \in S_I} \bar{p}_{a,b}^{(t_n)} \prod_{c \in S_O \setminus S_I} \left(1 - \prod_{d \in S_I} \bar{p}_{c,d}^{(t_n)} \right)$$

if $S_I \subseteq S_O$ and $P_N^{(1)}(S_I \rightarrow S_O; t_n) = 0$ otherwise, where $\bar{p}_{a,b}^{(t_n)} = 1 - p_{a,b}^{(t_n)}$.

Theorem 3.3 can hence be utilized to cast the indirect sharing version of our problem as a direct sharing one. In particular, an LP of the form described in (1) has to be solved, with the values of $p_k, \forall k \in [1 : K]$ being replaced with those obtained from Theorem 3.3. Note that these probabilities might not have the same symmetric structure as those of the direct sharing model³. However, the problem formulation and the algorithms designed in next sections are readily suitable for the indirect sharing case where the graph model is not necessarily symmetric. Thus, in the rest of the paper, for theoretical analysis we only consider the direct case. However, in Section 7, we assess the performance of our algorithms for both the direct and indirect cases.

4 CONNECTION TO SET-COVER PROBLEM

A Set-Cover (SC) problem is modeled as a bipartite graph $(\mathcal{S}, \mathcal{V}, \mathcal{E})$, with \mathcal{V} being the set of nodes (i.e., the *universe*), \mathcal{S} being a collection of sets whose union equals the universe and where an edge $e_{ij} \in \mathcal{E}$ exists between set $i \in \mathcal{S}$ and node $j \in \mathcal{V}$ if node j belongs to set i . An integer LP formulation of the SC problem then finds the optimal selection variables $x_i \in \{0, 1\}, \forall i \in [1 : N]$ to minimize the number of selected sets in \mathcal{S} while ‘covering’ all node in \mathcal{V} .

One can therefore think of the LP formulation in (1) as a relaxation of an integer LP, which models a variation of the SC problem. In this variation, there are two major differences: (i) the covering is performed on K bipartite graphs, each with a different adjacency matrix $\mathbf{A}^{(k)}, k \in [1 : K]$,

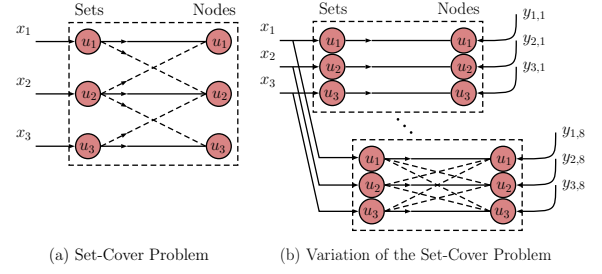


Figure 1. Set-cover problem and its variation.

and the same sets are selected to cover ‘all’ bipartite graphs; (ii) each node can be covered by either a selected set that contains it, or an ‘outside’ source. With reference to the LP in (1), the variables x_i are the selection variables of the sets, and the variables $y_{i,k}$ are the outside sources of user i in configuration k . An illustrative example is given in Figure 1. A conventional SC problem is shown in Figure 1(a), where the sets u_1 and u_3 contain nodes (u_1, u_2) and (u_2, u_3) , respectively, while set u_2 contains users $u_{[1:3]}$. The variables $x_{[1:3]}$ therefore determine which sets are selected for all the nodes to be covered. In this example, the set u_2 covers all the nodes. In our variation of the SC problem in Figure 1(b), there are 8 possible instances of bipartite graphs between 3 sets and 3 nodes, where the variables $x_{[1:3]}$ determine the selected sets that are used to simultaneously cover the users in *all* graphs, while the variables $y_{i,k}$ are used to cover the remaining users that were not covered by the selected sets.

The following theorem proves that indeed our LP formulation in (1) is closely related to the set-cover problem (see Appendix C for the proof).

Theorem 4.1. The optimal solution of the LP in (1) is lower bounded by the weighted sum of the outputs of K different LPs as follows. For $k \in [1 : K]$, the k -th LP is a relaxed SC problem over a bipartite graph with adjacency matrix $\mathbf{A}^{(k)}$. The output of the k -th LP is weighted by p_k .

5 POLYNOMIAL TIME APPROXIMATIONS

In this section, we propose heuristics that find an approximate solution for the LP in (1) in polynomial time.

Inverse Average Degree (IAD). Consider the symmetric direct case, where users meet pairwise with the same probability p . For this scenario, we expect that the bipartite graph has (in expectation) a constant degree of $p(N-1) + 1$, since each user, in average, meets the same number of people. The degree, in fact, captures the number of users met in that random realization; hence, each user meets (apart from herself) the remaining $N-1$ users with equal probability p .

In this case, a natural heuristic is to let each user download $\frac{1}{\mathbb{E}(C)}$, where C is a random variable corresponding to the number of people (including herself) a user meets. Figure 2 shows the optimal performance (solid lines), i.e., the solution of the LP in (1), and the performance of the caching strategy when each user downloads $\frac{1}{\mathbb{E}(C)}$ (dashed lines) for the symmetric case versus different values of p . It is evident from Figure 2, that such a choice of a caching strategy closely follows the performance of the optimal solution in symmetric scenarios. However, this approximation does not perform as well in the general (asymmetric) case. Consider,

2. The proof of Theorem 3.3 is based on simple counting techniques.

3. In the direct case, when user i meets user j with probability $p_{i,j}$, then user j meets user i with the same probability.

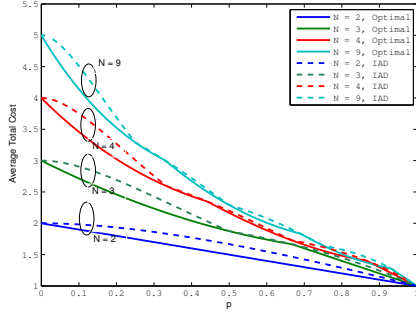


Figure 2. Optimal (solid lines) and IAD (dashed lines) average total cost.

for example, a ‘star’-like configuration, i.e., u_1 is highly connected to the other $N - 1$ users, while the other $N - 1$ users are only connected to u_1 . In this scenario the minimum (i.e., optimal) total average cost is approximately 1, achieved by letting u_1 download the whole file and then share it with the other $N - 1$ users. In contrast, if we force $u_i \in [1 : N]$ to download $\frac{1}{\mathbb{E}(C_i)}$ we would get that u_1 downloads $\frac{1}{N}$ (as she meets the others $N - 1$ members plus herself) and $u_j, j \neq 1$ downloads $\frac{1}{2}$ (as she only meets u_1 plus herself). This would imply a total cost of $(\frac{1}{N} + \frac{N-1}{2}) \geq 1, \forall N$ for the caching phase, which grows linearly with N and thus can be N -times worse than the optimal. This suggests that the optimal search might look like a ‘cover’: a set of nodes that enables to ‘reach’ and ‘convey’ information to all others. This is in line with the observations we previously made in Section 4.

Probabilistic Set-Cover (PSC). Building on this intuition, we propose another heuristic that seeks to find a form of a “fractional covering”, where the fraction that each user downloads is a ‘cover’ for the users she may meet. In the PSC problem [26], the covering constraint is replaced with a probabilistic one (i.e., the probability of covering all nodes is greater than a threshold). Here, we propose a variation of the PSC problem with an ‘average’ constraint.

We model the problem through a fully-connected bipartite graph $(\mathcal{U}, \mathcal{V}, \mathcal{E})$, where each edge $u_i - v_j, \forall u_i \in \mathcal{U}, v_j \in \mathcal{V}, (i, j) \in [1 : N]^2$ has an associated weight $p_{i,j}$, that represents how much on average u_i can cover v_j . We set $p_{i,i} = 1, \forall i \in [1 : N]$, and $p_{i,j} = p_{j,i}, \forall (i, j) \in [1 : N]^2, i \neq j$. The heuristic then seeks to associate fractional values x_i to the nodes in \mathcal{U} on the transmitting side, so that the sum of all x_i ’s is minimized, while each node in \mathcal{V} on the receiving side is covered, i.e., assured to receive (on average) the total amount. This is expressed through the following LP

$$\begin{aligned} \min_{\mathbf{x}} \quad & f^{\text{PSC}}(\mathbf{x}) = \sum_{i=1}^N x_i \\ \text{subject to} \quad & \mathbf{P}\mathbf{x} \geq \mathbf{1}_N, \mathbf{x} \geq \mathbf{0}_N, \end{aligned} \quad (4)$$

where \mathbf{P} is a matrix whose (i, j) -th entry (with $i \neq j$) is $p_{i,j}$ and with ones on the main diagonal. This is very similar to a fractional covering problem formulation, with the only difference that \mathbf{P} is not forced to be binary, but can have real components to express expectations.

The next theorem proves that, for the symmetric case, the optimal solution for the LP in (4) coincides with that of the IAD heuristic (see Appendix D for the proof).

Theorem 5.1. *For the symmetric scenario, the optimal solution for the LP in (4), denoted as \mathbf{x}^{PSC} , coincides with the IAD solution,*

Algorithm 1 AlgCov

Input Pairwise probability matrix: \mathbf{P}
Output AlgCov solution: \mathbf{x}^{Alg}
 Compute \mathbf{x}^{PSC} - the optimal solution of the LP in (4)
 Compute \mathbf{x}^{IAD} , with $x_i^{\text{IAD}} = 1/\mathbb{E}(C_i), \forall i \in [1 : N]$
if \mathbf{x}^{IAD} is feasible in (4) **then** $\mathbf{x}^{\text{Alg}} = \mathbf{x}^{\text{PSC}}$
else Compute $S^{\text{PSC}} = \mathbf{1}_N^T \mathbf{x}^{\text{PSC}}$ and $S^{\text{IAD}} = \mathbf{1}_N^T \mathbf{x}^{\text{IAD}}$
 if $S^{\text{IAD}} \leq S^{\text{PSC}}$ **then** $\mathbf{x}^{\text{Alg}} = \mathbf{x}^{\text{IAD}}$
 else $\mathbf{x}^{\text{Alg}} = \mathbf{x}^{\text{PSC}}$
end if
end if

denoted as \mathbf{x}^{IAD} , i.e., $\mathbf{x}^{\text{PSC}} = \mathbf{x}^{\text{IAD}} = \frac{1}{\mathbb{E}(C)} \mathbf{1}_N$ where $\mathbb{E}(C) = 1 + (N - 1)p$.

6 ALGCov ALGORITHM

In this section we present AlgCov, a simple heuristic algorithm that combines both approaches discussed in Section 5. AlgCov enables to calculate the fractions x_i in polynomial time, and achieves a performance close to that of the (exponentially complex) general LP in (1).

6.1 Motivation

To design an algorithm that combines the merits of both heuristics presented in Section 5, one might proceed as follows: (i) compute the solution \mathbf{x}^{PSC} of the PSC heuristic, (ii) compute the performance of this heuristic by plugging \mathbf{x}^{PSC} into the LP in (1) and by optimizing over \mathbf{y} to find the optimal cost for this solution. Then, repeat the same procedure for the IAD solution \mathbf{x}^{IAD} and finally choose the solution with the smallest cost.

Such a solution is, in theory, possible. However, the process of computing the cost of each heuristic involves solving an exponentially complex LP, prohibiting the applicability of the heuristic. The following theorem helps circumvent this complexity issue (see Appendix E for the proof).

Theorem 6.1. *Let \bar{f}^{Opt} and \bar{f}^{PSC} be the optimal values of the LPs in (1) and in (4), respectively. Then $\bar{f}^{\text{Opt}} \geq \bar{f}^{\text{PSC}}$.*

Theorem 6.1 provides a lower bound on the optimal value of the LP in (1), and consequently on the performance of the solution \mathbf{x}^{PSC} , i.e., $f^{\text{Opt}}(\mathbf{x}^{\text{PSC}}, \mathbf{y}^{\text{PSC}}) \geq \bar{f}^{\text{Opt}} \geq \bar{f}^{\text{PSC}}$, with \mathbf{y}^{PSC} being obtained by evaluating the LP in (1) while setting $\mathbf{x} = \mathbf{x}^{\text{PSC}}$. A fairly simple lower bound on the performance of \mathbf{x}^{IAD} is obtained by simply summing over the elements of the vector \mathbf{x}^{IAD} , i.e., $f^{\text{Opt}}(\mathbf{x}^{\text{IAD}}, \mathbf{y}^{\text{IAD}}) \geq \mathbf{1}_N^T \mathbf{x}^{\text{IAD}}$, with \mathbf{y}^{IAD} being obtained by evaluating the LP in (1) while setting $\mathbf{x} = \mathbf{x}^{\text{IAD}}$. As it is much simpler to compute these lower bounds, one can envisage to design an algorithm which, based on the lower bounds, selects one among the two heuristics described in Section 5.

6.2 Algorithm Description

AlgCov takes as input the probability matrix \mathbf{P} that contains the pairwise probabilities of meeting among users, and outputs the solution \mathbf{x}^{Alg} as a caching strategy. It first computes the two heuristic solutions, namely \mathbf{x}^{PSC} and \mathbf{x}^{IAD} , and then, as shown in Algorithm 1, selects one of them as output based on S^{PSC} and S^{IAD} , which in Theorem 6.1 we proved to be lower bounds on the actual performance of the heuristics.

6.3 Analytical performance

Symmetric case. In this setting, all pairs of members meet with equal probability. According to Theorem 5.1, both the IAD and the PSC heuristics provide the same solution, i.e., $\mathbf{x}^{\text{Alg}} = \mathbf{x}^{\text{PSC}} = \mathbf{x}^{\text{IAD}}$. By optimizing the objective function of the LP in (3) over y_i , $i \in [1 : N]$ with $\mathbf{x} = \mathbf{x}^{\text{Alg}}$, we obtain

$$f^{\text{Opt}}(\mathbf{x}^{\text{Alg}}) = \sum_{i=1}^N \left[1 - \frac{i}{1+(N-1)p} \right]^+ \cdot \left(\frac{N-1}{i-1} \right) N p^{i-1} (1-p)^{N-i} + \frac{N}{1+(N-1)p}, \quad (5)$$

which is an upper bound on the optimal performance, i.e., $\bar{f}^{\text{Opt}} \leq f^{\text{Opt}}(\mathbf{x}^{\text{Alg}})$. In order to provide a performance guarantee we need to understand how well $f^{\text{Opt}}(\mathbf{x}^{\text{Alg}})$ approximates the optimal solution of the LP in (3). To this end, we use the lower bound in Theorem 6.1. This lower bound, denoted as f^{LB} , implies that $\bar{f}^{\text{Opt}} \geq f^{\text{LB}}$. Using the structure of \mathbf{x}^{PSC} for the symmetric case in Theorem 5.1, the lower bound becomes

$$f^{\text{LB}} = \frac{N}{1+(N-1)p}. \quad (6)$$

By simply taking the difference between $f^{\text{Opt}}(\mathbf{x}^{\text{Alg}})$ in (5) and f^{LB} in (6) we obtain

$$G^{\text{sym}} \leq \sum_{i=1}^N \left[1 - \frac{i}{1+(N-1)p} \right]^+ \left(\frac{N-1}{i-1} \right) N p^{i-1} (1-p)^{N-i}.$$

The above gap result ensures us that, in the symmetric case, the output of AlgCov is always no more than G^{sym} above the optimal solution of the LP in (3). It is worth noting that G^{sym} is only function of the number of members N and of the probability p that users meet.

Remark 6.2. Through extensive numerical simulations, we observed that G^{sym} is maximum for $i = 1$, i.e., the probability p^* maximizing G^{sym} is $p^* = \frac{-N + \sqrt{5N^2 - 8N + 4}}{2(N-1)^2}$. By evaluating G^{sym} in p^* , we get a worst-case (greatest) gap of $G^{\text{sym}} \leq 0.25N$.

Asymmetric case. In this setting, different pairs of members meet with different probabilities. In this scenario, differently from the symmetric case analysed above, the LP in (4) does not seem to admit an easily computable closed-form solution. For this reason, we next show how the analysis drawn for the symmetric case can be extended to find a performance guarantee for the asymmetric case as well.

In the asymmetric case, an upper bound on the solution of AlgCov can be found by evaluating $f^{\text{Opt}}(\mathbf{x}^{\text{Alg}})$ in (5) in $p = p_m$, with $p_m = \min_{(i,j) \in [1:N]^2, i \neq j} \{p_{i,j}\}$. In other words, instead of considering different probabilities for different pairs, we set all of them to be equal to the minimum probability; this gives a solution which is always worse, i.e., greater than or equal to the optimal solution of AlgCov evaluated with the original (asymmetric) probability matrix.

Similarly, a lower bound on the optimal solution of the LP in (2) can be found by evaluating f^{LB} in (6) in $p = p_M$, with $p_M = \max_{(i,j) \in [1:N]^2, i \neq j} \{p_{i,j}\}$. Again, instead of considering different probabilities for different pairs, we set all of them to be equal to the maximum probability; this gives a solution which is always better, i.e., smaller than or

equal to the optimal solution of the LP in (2) evaluated with the original (asymmetric) probability matrix. Thus

$$G^{\text{asym}} \leq \sum_{i=1}^N \left[1 - \frac{i}{1+(N-1)p_m} \right]^+ \left(\frac{N-1}{i-1} \right) N p_m^{i-1} (1-p_m)^{N-i} + \frac{N(N-1)(p_M - p_m)}{[1+(N-1)p_m][1+(N-1)p_M]}.$$

This proves that in the asymmetric case, the output of AlgCov is always no more than G^{asym} above the optimal solution of the LP in (2). Similar to the symmetric case, also in this setting G^{asym} is only a function of the number of members N and of the probabilities p_m and p_M .

7 DATA-SET EVALUATION

In this section, we evaluate and compare the performance of our proposed solutions and algorithms using mobility traces that are obtained either from real-world experiments or via a human mobility trace synthesizer.

Performance Metrics and Comparisons: We are mainly interested in the performance of our proposed caching techniques in comparison to the conventional non-sharing solution. Specifically, we are interested in assessing the *average total cost* (total amount downloaded across the caching and post-sharing phases), averaged over the experiments. If each user simply downloads all data, this cost is N . Versus this, we compare the performance of:

- *Original Formulation and AlgCov:* We calculate the average probabilities from our dataset, feed these into the LP in (1) and into Algorithm 1 that assume Bernoulli distributions, and obtain the optimal and the AlgCov heuristic solutions, respectively. For each experiment, we then use these caching amounts, and follow the real meeting patterns recorded in the mobility traces to exchange data and download as needed in post-sharing phase. Finally, we calculate the actual total cost, averaged over the experiments.

- *1/N:* We evaluate the performance when each user caches $1/N$ of the data, independently of the meeting probabilities; this is a naive heuristic that does not fully exploit the opportunistic sharing possibilities.

- *CopCash:* We propose a modified version of the *cooperative sharing* algorithm originally proposed in [5], where we incorporate the concept of caching. Cooperative sharing takes advantage of the fact that nearby users, with a common demand, can collectively download the requested set of files. In addition, the proposed CopCash allows users to cache the received files, with the goal of exploiting next encounter opportunities to further share the data with other users. The scheme can be described as follows:

- 1) Whenever N users meet, each of them first downloads a fraction $1/N$ of the requested set of files, and then they share these parts among themselves through cost-free transmissions (e.g., Bluetooth).
- 2) If there exists a user (or a set of users) in the group who has already participated to a cooperative sharing instance, she directly shares what she has in her cache, i.e., what she obtained from previous meetings. In particular, she can share only what she has downloaded (direct sharing) or the whole set of files (indirect sharing).

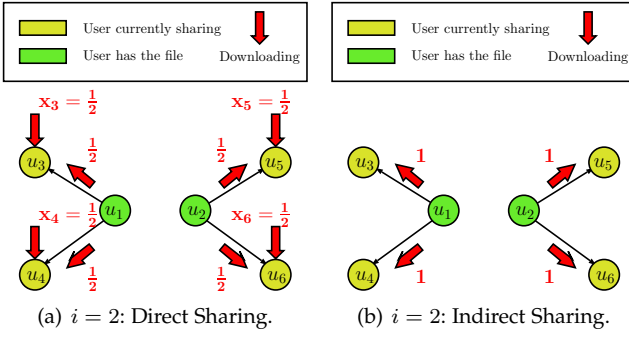


Figure 3. An illustrative example of CopCash.

- 3) The sharing procedure continues until the end of the period of duration t . At this point, if a user has participated in a previous sharing instance, she will have already obtained the set of files during that sharing instance. Otherwise, she will solely download the file set.

Consider the example in Figure 3. Suppose that, at time instant $i = 1$, u_1 and u_2 met; hence each of them downloaded $1/2$ of the file. Then, u_1 and u_2 exchanged the downloaded fractions, thus their demand was satisfied. At time instant $i = 2$, u_1 meets u_3 and u_4 , while u_2 meets u_5 and u_6 . In the case of direct sharing - see Figure 3(a) - u_1 (respectively, u_2) shares with u_3 and u_4 (respectively, u_5 and u_6) what she has personally downloaded from the server, i.e., $1/2$ of the file; at the end of the sharing period, $u_j, \forall j \in [3 : 6]$ downloads $1/2$ of the file from the server. With this, each user has to download $1/2$ of the file. In the case of indirect sharing - see Figure 3(b) - u_1 and u_2 share the whole set of files with the users they are connected to; in this case, $u_j, \forall j \in [3 : 6]$ does not need to download anything from the server.

- **Target-Set:** We assess the performance of the Target-Set heuristic proposed in [14] with $k = 1$, i.e., the server assigns one user the task to route the data to other users. We only show the performance of $k = 1$ since it is the case which incurs the smallest cost over the datasets that we consider.

Experiment Setup: We consider groups of size $N = 6$. In each experiment, we obtain the average performance of our algorithms by averaging over 50 *group trials*. For each group trial we pick a group of size 6 according to a specific selection criterion, and we compute the performance of the different heuristics for this group. In particular, we evaluate the performance in two different types of network, namely:

- 1) **Symmetric Configurations:** Users in the group have approximately the same expected number of users to meet among the group. Note that this is a relaxed requirement of symmetry with respect to the one used in Section 3 where all the users were assumed to meet with the exact same probability.
- 2) **Asymmetric Configurations:** Users in the group have different expected number of users to meet.

For each group, we define the *Expectation Deviation (ED)*: the difference between the maximum and the minimum expected number of encountered users, among all users, i.e., let \mathcal{S}_j be the set of N users belonging to group j , then $ED = \max_{i \in \mathcal{S}_j} \mathbb{E}(C_i) - \min_{i \in \mathcal{S}_j} \mathbb{E}(C_i)$. A group with high ED is more

Description:	Value:
Number of Users	75
Dates of Interest	Oct., Nov., Dec. - 2004
Hours of Interest	2 pm - 6 pm
Sharing Period	15 mins
Deadlines t	1 h, 2 h, 4 h
Number of Group Trials	50
N	6
$th_{asym}, th_{sym}, th_{max}$	1.3, 0.2, 1.2

Table 1
Experiment Parameters - MIT Reality Mining Dataset.

likely to have an asymmetric structure, while a group with small ED would have a symmetric structure. Our selection criterion is therefore the following: (a) for asymmetric configurations, we choose groups that have $ED \geq th_{asym}$, and (b) for symmetric configurations, we select groups that have $ED \leq th_{sym}$, while having $\max_{i \in \mathcal{S}_j} \mathbb{E}(C_i) \geq th_{max}$; th_{asym} , th_{sym} and th_{max} are decision parameters. For each experiment, these thresholds are set to values, which ensure the existence of the required number of groups.

We consider different *deadlines t* : the time period after which all users must individually have the whole set of files at their disposal. Intuitively, we expect that the longer the deadline is, the higher the number of sharing opportunities can be among the users within the same group and thus the smaller the average cost becomes. For each deadline, the duration of the whole experiment is divided into a number of *deadline trials*: for example, if the experiment is performed for a duration of 100 days, and we consider a duration of 4 hours in each day, then for a deadline of 2 hours, we have $\frac{100 \cdot 4}{2} = 200$ deadline trials.

7.1 MIT Reality Mining Dataset

We evaluate the performance of our proposed solutions and algorithms using the dataset from the MIT Reality Mining project [17]. Table 1 lists the values of all the parameters that we use in our experiment, described in the following.

This dataset includes the traces from 104 subjects affiliated to MIT - 75 of whom were in the Media Laboratory - in the period from September 2004 to June 2005. All subjects were provided with Nokia 6300 smart phones used to collect information such as the Bluetooth devices in the proximity logs. In our experiment, we utilize this information to capture the sharing opportunities among users. Each device was programmed to perform a Bluetooth device discovery approximately every 5 minutes, and store both the time instant at which the scan was performed, as well as the list of the MAC addresses of the devices found during the scan.

Assumptions: We say that two users are connected at a time instant, if there exists a scan (at that time instant) that was performed by any of the two users, in which the other user was found. We assume *instantaneous* sharing, i.e., if two users are connected at a time instant, then they can share their full cache contents. We justify this assumption in the following discussion. As specified in [17], Bluetooth discovery scans were performed and logged approximately every 5 minutes. However, this granularity in time was not always attainable since (i) the devices were highly asynchronous,

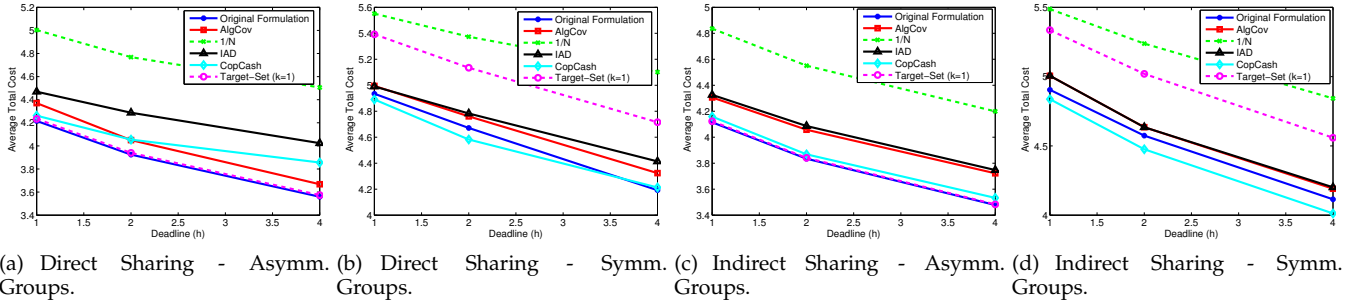


Figure 4. Experimental result obtained from the MIT Reality Mining dataset.

and (ii) some devices were powered off for a considerable amount of time. Because a non-negligible fraction of users experienced these irregularities, discarding their traces is not a suitable solution. Other solutions in the literature (for example, see [27]) utilize the IDs of the cell towers to which mobile devices are connected to infer proximity information. However, such approaches are too optimistic in assuming sharing opportunities, and hence are not suitable for our application. Our approach to deal with this highly irregular data was to consider the minimum sharing interval to be 15 minutes, i.e., two users are connected for an entire sharing interval if they are so at any time instant in that specific interval. Using the standard Bluetooth wireless transmission speed, this time period is sufficient to share approximately 2 GBs of data. Hence, for all practical purposes, it is reasonable to assume that any two connected users can share their full cache contents during that sharing interval.

For indirect sharing, we do not allow *intra-interval relaying*: users cannot indirectly share with other users within the same interval. We do, however, allow *inter-interval relaying*: indirect sharing can be performed across successive intervals. Our premise is that, while a 15-minute sharing interval is sufficient for one full cache content sharing, it might not be long enough to ensure more than one successful data exchange. This approach might severely limit the performance, i.e., a lower cost could be achieved by allowing intra-interval relaying.

Setup: We consider a period of three months from the academic year 2004/2005 in MIT, namely from October to December. We consider traces of only 75 users - labeled as affiliated to the Media Laboratory - during Monday, Tuesday and Wednesday. The reason for choosing these particular days is that we observed that, across the time period of interest, meetings occur most frequently in these days; thus, this represents a suitable period to assess the performance of all the solutions under consideration. We perform each experiment from 2 pm to 6 pm, and we consider deadlines of $t \in \{1, 2, 4\}$ hours. The thresholds for choosing groups are $th_{\text{asym}} = 1.3$, $th_{\text{sym}} = 0.2$ and $th_{\text{max}} = 1.2$. The reason behind this particular choice was to ensure the existence of 50 groups of 6 users in the duration of the experiment.

Experimental Results: Figure 4 shows the performance of different network structures (i.e., asymmetric and symmetric) for the direct and indirect sharing cases, respectively. From Figure 4, as expected, we observe that: (i) the average total cost decreases as the deadline increases; (ii) the average total cost in the indirect sharing case is less than the one

in the direct case, thanks to a higher number of sharing opportunities; (iii) using $1/N$ as a caching strategy performs the worst among all other schemes. This is because the $1/N$ scheme, differently from the other strategies, is not based on the meeting probabilities of the users.

Asymmetric Networks: Figure 4(a) and Figure 4(c) show the performance over asymmetric networks for the direct and indirect sharing cases, respectively. We note the following:

- Target-Set performs very close to the optimal scheme in both the direct and the indirect sharing cases. This is due to the asymmetric structure of the selected groups: one node is more likely to be connected to the other members of the group, and therefore the optimal solution would rely on that node to deliver the data to the whole group.

- AlgCov outperforms IAD in Figure 4(a), which indicates that AlgCov utilizes the solution that is generated from PSC. In contrast, AlgCov and the IAD strategy perform almost the same in Figure 4(c) which indicates that IAD outperforms PSC in this case. This justifies the merge between these two heuristics in the design of AlgCov.

Symmetric Networks: Figure 4(b) and Figure 4(d) show the performance of the different schemes over symmetric networks for the direct and the indirect sharing cases, respectively. Observations are similar to those drawn for the asymmetric case. However, one major observation is that Target-Set, differently from asymmetric groups, poorly performs. This is a direct consequence of the symmetric structure of the selected group: in a symmetric group, an optimal sharing strategy would equally distribute the caching and sharing efforts among all members within the group; in contrast, Target-Set selects only one member who has the task of caching and sharing the data for the group.

Remark 7.1. One might argue that CopCash has an inherent advantage over the other caching strategies since it does not need the genie-aided information of the pairwise meeting probabilities. However, this information is not hard to obtain in a realistic scenario. For example, although being out of the scope of this work, one can think of modifying AlgCov, by including a learning module. With this and by exploiting the regular mobility behavior of the users, the probabilities can be estimated as reportedly done in the literature (see [15], [27]).

Remark 7.2. CopCash performs closely to our proposed solution. One can thus draw a premature conclusion that pre-caching does not bring significant benefits with respect to opportunistically exploiting sharing opportunities, as CopCash does. This is true when the meeting probabilities are small, as in the MIT Reality Mining dataset. However, as shown next, pre-caching solutions

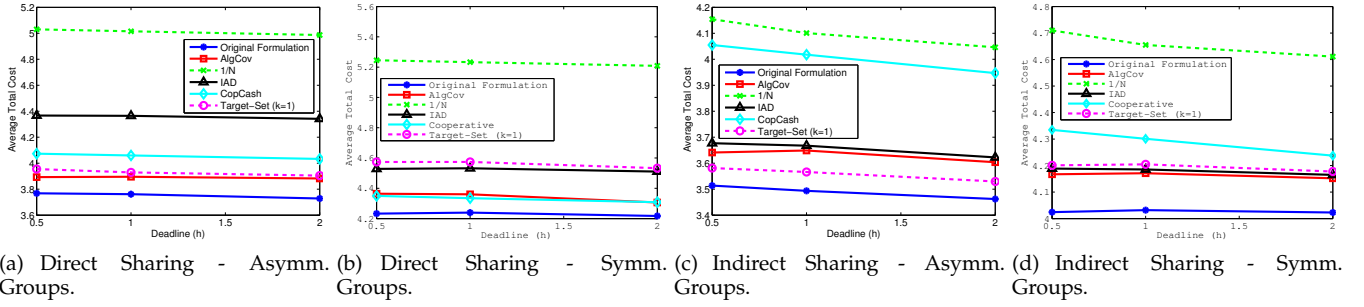


Figure 5. Experimental results obtained from the synthesized Infocom-2005 trace.

Description:	Value:
Number of Users	300
Duration of Experiment	3 (Infocom-2005) and 11 (Cambridge-2006) days
Sharing Period	10 mins
Deadlines t	0.5 h, 1 h, 2 h
Number of Group Trials	50
N	6
$th_{\text{asym}}, th_{\text{sym}}, th_{\text{max}}$	3.7, 0.2, 1.2

Table 2
Experiment Parameters - Infocom-2005 and Cambridge-2006.

outperform opportunistic sharing approaches when the users are moderately/highly connected.

7.2 SWIM-Based Results

We here evaluate the performance of our algorithms over mobility traces synthesized using the SWIM model. SWIM [18] is a human mobility model that is used to synthesize mobility traces based on the users social behavior. Traces are generated in the form of *events*: the exact time at which two users meet/leave. Thus, the trace files consist of a chronological series of meeting/leaving events among the users involved in the generation of the trace. We use a synthesized version of two existing traces, namely Infocom-2005 and Cambridge-2006. These traces were obtained through experiments conducted in the IEEE INFOCOM 2005 conference and in Cambridge in 2006, respectively (see [19], [20] for more details). The synthesized versions of these traces include a greater number of nodes (with the same spatial density) than the original ones, which is the main reason behind our choice of the synthesized traces.

Assumptions: We consider the sharing interval to be 10 minutes. We say that two users successfully exchange their cache contents if they are in contact for at least 85% of the interval. Similarly to Section 7.1, in the indirect sharing we only allow inter-interval relaying.

Setup: We perform each experiment over the traces from 300 virtual users during the entire duration of the trace (3 days for Infocom-2005 and 11 days for Cambridge-2006). The deadlines that we consider are of $t = 0.5$ hour, $t = 1$ hour and $t = 2$ hours. The thresholds for choosing groups are $th_{\text{asym}} = 3.7$, $th_{\text{sym}} = 0.2$ and $th_{\text{max}} = 1.2$. The reason behind this particular choice was to ensure the existence of 50 groups of 6 users for all the days of the experiment. Table 2 lists the values of all the parameters of the experiments.

Experimental Results: We assess the performance of our algorithms on the Infocom-2005 (Figure 5) and Cambridge-2006 (Figure 6) mobility traces. Similar conclusions to those in Section 7.1 can be drawn. In particular: (i) the average total cost decreases as the deadline increases; (ii) the average total cost incurred in the indirect sharing case is less than the one in the direct counterpart; (iii) the caching strategy $1/N$ shows the worst performance among the different schemes; (iv) Target-Set performs close to the optimal in asymmetric configurations. However, differently from Section 7.1, in most of the cases CopCash poorly performs with respect to other solutions. The reason is that the mobility traces of both Infocom-2005 and Cambridge-2006 show a relatively high frequency of meetings among users, which is a distinct feature with respect to the MIT Reality Mining dataset.

8 CONCLUSIONS

We here motivated, proposed, analysed, and experimentally evaluated AlgCov, a simple low-complexity algorithm for social caching, that uses pre-caching in anticipation of encounter opportunities to minimize the required download bandwidth. We derived formal LP formulations and presented a worst-case analytical performance gap. We numerically evaluated the performance of the proposed solutions on (i) the mobility traces obtained from the MIT Reality Mining data set, and (ii) two mobility traces that were synthesized using the SWIM mobility model. AlgCov achieves a performance which is close to the optimal and, in some configurations, it outperforms existing solutions, such as the Target-Set. AlgCov makes the case that, even in the presence of random encounters, using simple algorithms for pre-caching can significantly reduce bandwidth usage.

APPENDIX A PROOF OF THEOREM 3.1

The key observation is to notice that the constraints in the LP in (1) can be written in the form $1 - \pi_v \mathbf{x} \leq y_{i,k}$, $\forall y_{i,k} \in \mathcal{Y}^{(v)}$, where $\mathcal{Y}^{(v)} = \{y_{i,k} | 1 - \pi_v \mathbf{x} \leq y_{i,k} \text{ is a constraint}\}$. Since all the constraints of the type $\mathbf{1}_N - \mathbf{A}^{(k)} \mathbf{x} \leq \mathbf{y}_k$ in the LP in (1) can be replaced with $1 - \pi_v \mathbf{x} \leq y_{i,k}$, the optimal solution would make all $y_{i,k} \in \mathcal{Y}^{(v)}$ equal, as proved in Lemma A.1.

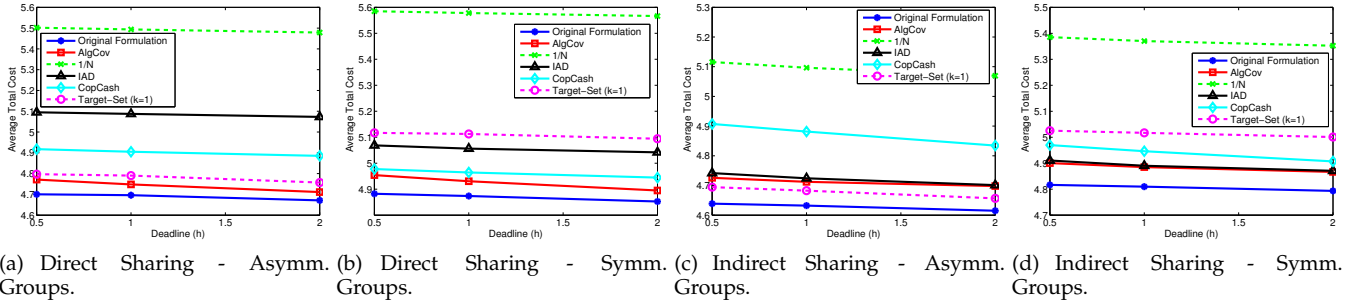


Figure 6. results obtained from the synthesized Cambridge-2006 trace.

Lemma A.1. Let $(\hat{\mathbf{x}}, \hat{\mathbf{y}})$ be an optimal solution for the LP in (1). Then $\hat{y}_{i,k} = y_v, \forall \hat{y}_{i,k} \in \mathcal{Y}^{(v)}$. Thus,

$$f^{\text{Opt}}(\hat{\mathbf{x}}, \hat{\mathbf{y}}) = \sum_{v=1}^{2^N-1} y_v \sum_{i=1}^N \sum_{\substack{k=1, \\ y_{i,k} \in \mathcal{Y}^{(v)}}}^K p_k + \mathbf{1}_N^T \hat{\mathbf{x}}.$$

We next prove the result in Lemma A.1. Without loss of generality, assume that $\hat{y}_{i,k} = y_v, \forall \hat{y}_{i,k} \in \mathcal{Y}^{(v)} \setminus \bar{y}$ and $\bar{y} = y_v + \Delta$ where $\Delta \geq 0$. Then, since this is a feasible point, Δ can be driven down to zero without violating the feasibility conditions, and consequently reducing the optimal value of the objective function; thus, we have a contradiction. The same argument can be extended to the case where more than one $\hat{y}_{i,k}$ is different from y_v .

With $y_{i,k} = y_v, \forall y_{i,k} \in \mathcal{Y}^{(v)}$ in $f^{\text{Opt}}(\mathbf{x}, \mathbf{y})$ in (1), we get

$$\begin{aligned} \sum_{k=1}^K p_k \sum_{i=1}^N y_{i,k} &= \sum_{k=1}^K p_k \sum_{i=1}^N \sum_{v=1}^{2^N-1} y_{i,k} \mathbb{1}_{\{y_{i,k} \in \mathcal{Y}^{(v)}\}} \\ &= \sum_{v=1}^{2^N-1} y_v \sum_{i=1}^N \sum_{k=1}^K p_k \mathbb{1}_{\{y_{i,k} \in \mathcal{Y}^{(v)}\}}. \end{aligned}$$

This concludes the proof of Lemma A.1.

Notice that, by our definition in Theorem 3.1, we have

$$\sum_{i=1}^N \sum_{k=1}^K p_k \mathbb{1}_{\{y_{i,k} \in \mathcal{Y}^{(v)}\}} = \sum_{u \in \mathcal{S}_v} \Pr(u \rightarrow \mathcal{S}_v).$$

We now use the result in Lemma A.1 to prove Theorem 3.1, i.e., the equivalence of the LPs in (1) and in (2).

Part 1. Let $(\mathbf{x}^1, \mathbf{y}^1)$ be an optimal solution for the LP in (1), which follows the structure described in Lemma A.1. For $v \in [1 : 2^N - 1]$, let y_v^1 be the value where, for each $y_{i,k}^1 \in \mathcal{Y}^{(v)}$, $y_{i,k}^1 = y_v^1$. Then, one can construct a feasible solution $(\mathbf{x}^2, \mathbf{y}^2)$ for the LP in (2) as follows: (i) set $\mathbf{x}^2 = \mathbf{x}^1$; (ii) let y_v^2 be an element of \mathbf{y}^2 that corresponds to a constraint of the form $1 - \pi_v \mathbf{x}^2 \leq y_v^2$ in the LP in (2), then set $y_v^2 = y_v^1$. By doing so, the constraints of the LP in (2) are satisfied. Moreover, with Lemma A.1, the objective functions of both problems are equal, when evaluated at the described points.

Part 2. Let $(\mathbf{x}^2, \mathbf{y}^2)$ be an optimal solution for the LP in (2). Then one can construct a feasible solution $(\mathbf{x}^1, \mathbf{y}^1)$ for the LP in (1) as follows: (i) set $\mathbf{x}^1 = \mathbf{x}^2$; (ii) $\forall y_{i,k}^1 \in \mathcal{Y}^{(v)}$, set $y_{i,k}^1 = y_v^2$. By doing so, the constraints of the LP in (1) are guaranteed to be satisfied. Moreover, with Lemma A.1, the objective functions of both problems will be equal, when evaluated at the described points. This concludes the proof.

APPENDIX B PROOF OF THEOREM 3.2

We prove the equivalence of the LP in (1) and the LP in (3) by means of the following lemma.

Lemma B.1. Let $(\hat{\mathbf{x}}, \hat{\mathbf{y}})$ be the optimal solution for the LP in (1). Then, by assuming a symmetric model

- 1) $\hat{\mathbf{x}} = \hat{x} \mathbf{1}_N$, with $\hat{x} \in [0, 1]$;
- 2) For $m \in [1 : N]$, let $\Pi^{(m)} = \{(i, k) | [1 - m\hat{x}]^+ \leq \hat{y}_{i,k} \leq [1 - (m-1)\hat{x}]^+\}$ with $i \in [1 : N]$, $k \in [1 : K]$, then $\hat{y}_{i,k} = \hat{y}_m, \forall (i, k) \in \Pi^{(m)}$, with $\hat{y}_m = [1 - m\hat{x}]^+$.

Moreover, with reference to $f^{\text{Opt}}(\mathbf{x}, \mathbf{y})$ in (1), we get

$$\sum_{k=1}^K p_k \sum_{i=1}^N \hat{y}_{i,k} = \sum_{i=1}^N \hat{y}_i \binom{N-1}{i-1} N p^{i-1} (1-p)^{N-i}.$$

We now prove Lemma B.1 in three steps.

Step 1. We prove that $\mathbf{x} = x \mathbf{1}_N$ and $y_{i,k} = y_m \forall (i, k) \in \Pi^{(m)}$, $m \in [1 : N]$ is a feasible solution for the LP in (1). Assume a feasible solution consists of $\mathbf{x} = x \mathbf{1}_N$. Then, with $y_{i,k} = y_m \forall (i, k) \in \Pi^{(m)}$, $m \in [1 : N]$ so that $y_m \geq [1 - mx]^+$, we get a feasible solution of the required form.

Step 2. Assume that an optimal solution has $\hat{\mathbf{x}} = \hat{x} \mathbf{1}_N$. We prove, by contradiction, that this implies $\hat{y}_{i,k} = \hat{y}_m \forall (i, k) \in \Pi^{(m)}$. We use similar steps as in the proof of Lemma A.1. Without loss of generality, assume that $\hat{y}_{i,k} = \hat{y}_m \forall (i, k) \in \Pi^{(m)} \setminus (\bar{i}, \bar{k})$ and $\hat{y}_{\bar{i}, \bar{k}} = \hat{y}_m + \Delta$, where $\Delta \geq 0$. Since this point is feasible, then Δ can be driven down to zero without having violated the feasibility conditions; this operation (i.e., setting $\Delta = 0$) also implies a reduction in the optimal value of the objective function; thus we have a contradiction.

Step 3. We prove that, for the symmetric model, an optimal solution of the form $\hat{\mathbf{x}} = \hat{x} \mathbf{1}_N$ and $\hat{y}_{i,k} = \hat{y}_m \forall (i, k) \in \Pi^{(m)} \forall m \in [1 : N]$ exists. Without loss of generality, assume that $(\tilde{\mathbf{x}}, \tilde{\mathbf{y}})$ is an optimal solution of the form $\tilde{\mathbf{x}} = [x, \dots, x, x + \Delta]^T$, where $0 \leq \Delta < x^4$. We show that $\hat{\mathbf{x}} = x + \frac{\Delta}{N} \mathbf{1}_N$ gives a smaller value for the objective function of the LP in (1), i.e., $f^{\text{Opt}}(\tilde{\mathbf{x}}, \tilde{\mathbf{y}}) - f^{\text{Opt}}(\hat{\mathbf{x}}, \hat{\mathbf{y}}) =$

4. This assumption is not necessary and is made only to simplify the analysis.

$\sum_{k=1}^K p_k \sum_{i=1}^N \tilde{y}_{i,k} - \sum_{k=1}^K p_k \sum_{i=1}^N \hat{y}_{i,k} \geq 0$. We start by noticing that, by using the symmetric model in $f^{\text{Opt}}(\mathbf{x}, \mathbf{y})$ in (1), we get

$$\begin{aligned} \sum_{k=1}^K p_k \sum_{i=1}^N y_{i,k} &= p_1 \left(\sum_{i=1}^N y_{i,1} \right) + \dots + p_K \left(\sum_{i=1}^N y_{i,K} \right) \\ &= \sum_{j=0}^{\frac{N(N-1)}{2}} B(p, j, N) \sum_{k=1}^K \sum_{i=1}^N y_{i,k} \mathbb{1}_{\{p_k=B(p,j,N)\}}, \end{aligned} \quad (7)$$

where the last equality follows by noticing that each of the $p_k, k \in [1 : K]$ is equal to a term of the type $B(p, j, N) = p^j (1-p)^{\frac{N(N-1)}{2}-j}$ for some $j \in [0 : \frac{N(N-1)}{2}]$ and by swapping the order of the summations.

We next evaluate $f^{\text{Opt}}(\tilde{\mathbf{x}}, \tilde{\mathbf{y}})$ and $f^{\text{Opt}}(\hat{\mathbf{x}}, \hat{\mathbf{y}})$ separately.

Evaluation of $f^{\text{Opt}}(\tilde{\mathbf{x}}, \tilde{\mathbf{y}})$: We define

$$\begin{aligned} \tilde{\Pi}_{\Delta}^{(m,j)} &= \{(i, k) | p_k = B(p, j, N), \\ &\quad [1 - mx - \Delta]^+ \leq \tilde{y}_{i,k} < [1 - mx]^+ \}, \\ \tilde{\Pi}_{\Delta, \dagger}^{(m,j)} &= \{(i, k) | p_k = B(p, j, N), \\ &\quad [1 - mx]^+ \leq \tilde{y}_{i,k} < [1 - (m-1)x - \Delta]^+ \}. \end{aligned}$$

As $\tilde{\mathbf{x}}$ is fixed, the optimal solution would yield $\tilde{\mathbf{y}}$ to be as small as possible, while preserving feasibility. Thus, $\forall (i, k) \in \tilde{\Pi}_{\Delta}^{(m,j)}, \tilde{y}_{i,k} = [1 - mx - \Delta]^+$, and $\forall (i, k) \in \tilde{\Pi}_{\Delta, \dagger}^{(m,j)}, \tilde{y}_{i,k} = [1 - mx]^+$. By noticing that the sets $\tilde{\Pi}_{\Delta}^{(m,j)}$ and $\tilde{\Pi}_{\Delta, \dagger}^{(m,j)}$ are disjoint $\forall m \in [1 : N]$ and $\forall j \in [0 : \frac{N(N-1)}{2}]$ and contain all elements of $\tilde{\mathbf{y}}$, we can rewrite (7) as

$$\sum_{k=1}^K p_k \sum_{i=1}^N \tilde{y}_{i,k} = \sum_{j=0}^{\frac{N(N-1)}{2}} B(p, j, N) \sum_{m=1}^N \left[\sum_{w \in \tilde{\Pi}_{\Delta}^{(m,j)}} \tilde{y}_w + \sum_{w \in \tilde{\Pi}_{\Delta, \dagger}^{(m,j)}} \tilde{y}_w \right]. \quad (8)$$

Let $Q_N^{m,j}(a, b) = a^{\binom{N-1}{b-1}} \left(\frac{N(N-1)}{2} - m + 1 \right)$. Then by means of counting techniques, one can see that the number of elements of $\tilde{\mathbf{y}}$ that belong to a constraint of the type $1 - mx - \Delta$ is $|\tilde{\Pi}_{\Delta}^{(m,j)}| = Q_N^{m,j}(m, m)$, while the number of elements of $\tilde{\mathbf{y}}$ that belong to a constraint of the type $1 - mx$ is $|\tilde{\Pi}_{\Delta, \dagger}^{(m,j)}| = Q_N^{m,j}(N - m, m)$. With this we can rewrite (8) as in (9) at the top of the next page, where $g_1 \in \mathbb{Z}^+$, respectively $g_2 \in \mathbb{Z}^+$ which ensures that $[1 - \ell x - \Delta]^+ = 1 - \ell x - \Delta, \forall \ell \in [1 : g_1]$, respectively $[1 - \ell x]^+ = 1 - \ell x, \forall \ell \in [1 : g_2]$.

Evaluation of $f^{\text{Opt}}(\hat{\mathbf{x}}, \hat{\mathbf{y}})$: We define

$$\begin{aligned} \hat{\Pi}_{\Delta}^{(m,j)} &= \{(i, k) | p_k = B(p, j, N), \\ &\quad \left[1 - mx - \frac{m}{N} \Delta \right]^+ \leq \hat{y}_{i,k} < \left[1 - (m-1)x - \frac{m-1}{N} \Delta \right]^+ \}. \end{aligned}$$

Thus, similarly to the case of $(\tilde{\mathbf{x}}, \tilde{\mathbf{y}})$, the optimal solution would yield $\forall (i, k) \in \hat{\Pi}_{\Delta}^{(m,j)}, \hat{y}_{i,k} = \left[1 - mx - \frac{m}{N} \Delta \right]^+$ and we can rewrite (7) as

$$\sum_{k=1}^K p_k \sum_{i=1}^N \hat{y}_{i,k} = \sum_{j=0}^{\frac{N(N-1)}{2}} B(p, j, N) \cdot \sum_{m=1}^N \sum_{w \in \hat{\Pi}_{\Delta}^{(m,j)}} \hat{y}_w. \quad (11)$$

Similar to $(\tilde{\mathbf{x}}, \tilde{\mathbf{y}})$, one can see that the number of elements of $\hat{\mathbf{y}}$ that belong to a constraint of the type $1 - mx - \frac{m}{N} \Delta$ is $|\hat{\Pi}_{\Delta}^{(m,j)}| = Q_N^{m,j}(N, m)$. With this we have that (11) can be rewritten as (10) at the top of the next page, where $g_3 \in \mathbb{Z}^+$ ensures that $[1 - \ell x - \frac{\ell}{N} \Delta]^+ = 1 - \ell x - \frac{\ell}{N} \Delta, \forall \ell \in [1 : g_3]$.

Depending on x and Δ , one can distinguish 4 cases that might occur, which are next analyzed. For each case, we identify the values of $g_{[1:3]}$ and show that $f^{\text{Opt}}(\tilde{\mathbf{x}}, \tilde{\mathbf{y}}) - f^{\text{Opt}}(\hat{\mathbf{x}}, \hat{\mathbf{y}}) \geq 0$, thus proving optimality of the pair $(\tilde{\mathbf{x}}, \tilde{\mathbf{y}})$.

Case 1: $1 - (n+1)x > 0, n \in [1 : N]$; here, we have $g_1 = g_3 = n$ and $g_2 = n+1$ in (9) and (10) at the top of the next page. In that case we get $f^{\text{Opt}}(\tilde{\mathbf{x}}, \tilde{\mathbf{y}}) - f^{\text{Opt}}(\hat{\mathbf{x}}, \hat{\mathbf{y}}) \geq 0$.

Case 2: $1 - (n+1)x \leq 0$ and $1 - nx - \Delta > 0, n \in [1 : N]$; here, we have $g_1 = g_2 = g_3 = n$ in (9) and (10) at the top of the next page. In that case we get $f^{\text{Opt}}(\tilde{\mathbf{x}}, \tilde{\mathbf{y}}) - f^{\text{Opt}}(\hat{\mathbf{x}}, \hat{\mathbf{y}}) = 0$.

Case 3: $1 - nx - \Delta \leq 0$ and $1 - nx - \frac{n}{N} \Delta > 0, n \in [1 : N]$; here, we have $g_1 = n-1$ and $g_2 = g_3 = n$ in (9) and (10) at the top of the next page. In that case we get $f^{\text{Opt}}(\tilde{\mathbf{x}}, \tilde{\mathbf{y}}) - f^{\text{Opt}}(\hat{\mathbf{x}}, \hat{\mathbf{y}}) \geq 0$.

Case 4: $1 - nx - \frac{n}{N} \Delta \leq 0$ and $1 - nx > 0, n \in [1 : N]$; here, we have $g_1 = g_3 = n-1$ and $g_2 = n$ in (9) and (10) at the top of the page. In that case we get $f^{\text{Opt}}(\tilde{\mathbf{x}}, \tilde{\mathbf{y}}) - f^{\text{Opt}}(\hat{\mathbf{x}}, \hat{\mathbf{y}}) \geq 0$.

Remark B.2. The proof above generalizes to the case when $\tilde{\mathbf{x}}$ has general components. The idea is to order $\tilde{\mathbf{x}}$ in ascending order and to rewrite it as $\tilde{\mathbf{x}} = x_m + [0, \Delta_2, \dots, \Delta_N]^T$, where $x_m = \min_i \tilde{x}_i$ and $\Delta_i, i \in [1 : N]$ is the difference between the i -th component of the ordered $\tilde{\mathbf{x}}$ and x_m . Then, the above method is applied $N-1$ times as follows. At step $k \in [1 : N-1]$, Δ_{k+1} is equally shared among the N components of $\tilde{\mathbf{x}}$; this, as proved above, brings to a reduction of the objective function. At the end of the $N-1$ steps the optimal $\hat{\mathbf{x}}$ is of the form $\hat{\mathbf{x}} = \mathbf{1}_N \hat{x}$.

According to the structure of $\hat{\mathbf{x}}$ in Step 2, the optimal $\hat{\mathbf{y}}$ has components of the form stated in Lemma B.1 - item 2).

One can see that with $f^{\text{Opt}}(\hat{\mathbf{x}}, \hat{\mathbf{y}})$ in (1), we have

$$\sum_{k=1}^K p_k \sum_{i=1}^N \hat{y}_{i,k} = \sum_{i=1}^N \hat{y}_i \binom{N-1}{i-1} N p^{i-1} (1-p)^{N-i},$$

which follows by noting that the probability of having i people meeting is $\binom{N-1}{i-1} p^{i-1} (1-p)^{N-i}$ and that this event happens once for every user. This completes the proof of Lemma B.1.

Using Lemma B.1, one can prove the equivalence of the LPs in (1) and (3) using similar arguments as in Appendix A. This concludes the proof.

APPENDIX C PROOF OF THEOREM 4.1

Let \bar{f}^{Opt} be the optimal solution for the LP in (1). Then, the LP in (1) can be equivalently written as

$$\bar{f}^{\text{Opt}} = \min_{(\mathbf{x}, \mathbf{y}) \in \mathcal{F}} \sum_{k=1}^K p_k \sum_{i=1}^N y_{i,k} + \mathbf{1}_N^T \mathbf{x},$$

$$\begin{aligned}
\sum_{k=1}^K p_k \sum_{i=1}^N \tilde{y}_{i,k} &= \sum_{j=0}^{\frac{N(N-1)}{2}} B(p, j, N) \left[\sum_{m=1}^N [1 - mx - \Delta]^+ Q_N^{m,j}(m, m) + \sum_{m=1}^N [1 - mx]^+ Q_N^{m,j}(N - m, m) \right] \\
&= \sum_{j=0}^{\frac{N(N-1)}{2}} B(p, j, N) \left[\sum_{\ell=1}^{g_1} (1 - \ell x - \Delta) Q_N^{\ell,j}(\ell, \ell) + \sum_{\ell=1}^{g_2} (1 - \ell x) Q_N^{\ell,j}(N - \ell, \ell) \right], \quad (9) \\
\sum_{k=1}^K p_k \sum_{i=1}^N \hat{y}_{i,k} &= \sum_{j=0}^{\frac{N(N-1)}{2}} B(p, j, N) \sum_{m=1}^N \left[1 - mx - \frac{m}{N} \Delta \right]^+ Q_N^{m,j}(N, m) = \sum_{j=0}^{\frac{N(N-1)}{2}} B(p, j, N) \sum_{\ell=1}^{g_3} \left(1 - \ell x - \frac{\ell}{N} \Delta \right) Q_N^{\ell,j}(N, \ell). \quad (10)
\end{aligned}$$

where $\mathcal{F} = \{(\mathbf{x}, \mathbf{y}) \mid \mathbf{x} \geq \mathbf{0}_N, \mathbf{y} \geq \mathbf{0}_{N \times K}, \mathbf{1}_N - \mathbf{A}^{(k)} \mathbf{x} \leq \mathbf{y}_k, \forall k \in [1 : K]\}$. The next series of inequalities hold

$$\begin{aligned}
\bar{f}^{\text{Opt}} &= \min_{(\mathbf{x}, \mathbf{y}) \in \mathcal{F}} \sum_{k=1}^K p_k \sum_{i=1}^N y_{i,k} + \mathbf{1}_N^T \mathbf{x} \\
&= \min_{(\mathbf{x}, \mathbf{y}) \in \mathcal{F}} \sum_{k=1}^K p_k \sum_{i=1}^N y_{i,k} + \sum_{k=1}^K p_k \mathbf{1}_N^T \mathbf{x} \\
&\stackrel{(a)}{\geq} \sum_{k=1}^K p_k \min_{(\mathbf{x}, \mathbf{y}) \in \mathcal{F}} \left(\sum_{i=1}^N y_{i,k} + \mathbf{1}_N^T \mathbf{x} \right) = \sum_{k=1}^K p_k f(k),
\end{aligned}$$

where (a) is due to Jensen's inequality. Thus,

$$\begin{aligned}
f(k) &= \min_{\mathbf{x}, \mathbf{y}} \sum_{i=1}^N y_{i,k} + \mathbf{1}_N^T \mathbf{x} \\
&\text{subject to } \mathbf{x} \geq \mathbf{0}_N, \mathbf{y} \geq \mathbf{0}_{N \times K}, \\
&\quad \mathbf{1}_N - \mathbf{A}^{(v)} \mathbf{x} \leq \mathbf{y}_v, \forall v \in [1 : K].
\end{aligned} \quad (12)$$

Consider the set of constraints in (12) $\forall v \in [1 : K]$ such that $v \neq k$, which can be written as $\mathbf{A}^{(v)} \mathbf{x} \geq \mathbf{1}_N - \mathbf{y}_v \geq -\infty$, since \mathbf{y}_v does not directly affect the objective function. This makes these constraints trivial, i.e., (12) becomes

$$\begin{aligned}
f(k) &= \min_{\mathbf{x}, \mathbf{y}_k} \mathbf{1}_N^T \mathbf{y}_k + \mathbf{1}_N^T \mathbf{x} \\
&\text{subject to } \mathbf{x} \geq \mathbf{0}_N, \mathbf{y}_k \geq \mathbf{0}_N, \\
&\quad \mathbf{y}_k + \mathbf{A}^{(k)} \mathbf{x} \geq \mathbf{1}_N.
\end{aligned} \quad (13)$$

For the problem in (13), we prove that, for any $k \in [1 : K]$, an optimal solution of the form $\mathbf{y}_k = \mathbf{0}_N$ always exists. Assume this is not true, i.e., $\exists (\mathbf{x}^*, \mathbf{y}_k^*)$ such that $y_{i,k} > 0$, for $i \in \mathcal{S}$, where $\mathcal{S} \subseteq [1 : N]$. Then consider the point $(\hat{\mathbf{x}}, \hat{\mathbf{y}}_k)$. Note that $\mathbf{A}^{(k)}$ has ones on the diagonal. Then, by letting $\hat{\mathbf{y}}_k = \mathbf{0}_N$ and $\hat{\mathbf{x}} = \mathbf{x}^* + \mathbf{y}_k^*$ we get another feasible point with the same objective function. Thus, the problem in (13) becomes

$$\begin{aligned}
f(k) &= \min_{\mathbf{x}} \mathbf{1}_N^T \mathbf{x} \\
&\text{subject to } \mathbf{x} \geq \mathbf{0}_N, \mathbf{A}^{(k)} \mathbf{x} \geq \mathbf{1}_N.
\end{aligned} \quad (14)$$

The problem in (14) is the LP relaxation of the SC problem on a bipartite graph with adjacency matrix $\mathbf{A}^{(k)}$. This concludes the proof.

APPENDIX D

PROOF OF THEOREM 5.1

We start by proving that $\mathbf{x}^{\text{PSC}} = x \mathbf{1}_N$. Assume that \mathbf{x}' is the optimal solution, with $k = \min_{i \in [1 : N]} \{\mathbf{x}'_i\}$; without loss of generality, $\mathbf{x}' = k \mathbf{1}_N + [\Delta_{[1 : N-1]}, 0]^T$, where $\Delta_i \geq 0 \forall i =$

$[1 : N-1]$. With this, we get that $f^{\text{PSC}}(\mathbf{x}') = Nk + \sum_{i=1}^{N-1} \Delta_i$, and \mathbf{x}' satisfies the constraints of the LP in (4) that are

$$\begin{aligned}
k + (N-1)pk + \Delta_i + p \sum_{j=1, j \neq i}^{N-1} \Delta_j &\geq 1, \forall i \in [1 : N-1] \\
k + (N-1)pk + p \sum_{j=1}^{N-1} \Delta_j &\geq 1.
\end{aligned} \quad (15)$$

It is clear that the first set of constraints is always redundant, as the second one is tighter. Now consider $\mathbf{x}^{\text{PSC}} = x \mathbf{1}_N$, with $x = k + \frac{1}{N} \sum_{j=1}^{N-1} \Delta_j$; it is not difficult to see that $f^{\text{PSC}}(\mathbf{x}^{\text{PSC}}) = f^{\text{PSC}}(\mathbf{x}')$. To complete the proof that $\mathbf{x}^{\text{PSC}} = x \mathbf{1}_N$ we need to show that such a point is feasible. The constraints of the LP in (4) when evaluated at \mathbf{x}^{PSC} become

$$k + (N-1)pk + \frac{1}{N} (1 + (N-1)p) \sum_{j=1}^{N-1} \Delta_j \geq 1,$$

thus \mathbf{x}^{PSC} is a feasible solution as this constraint is always satisfied if the second constraint in (15) holds since $\frac{1}{N} (1 + (N-1)p) \geq p$. By enforcing this solution into the LP in (4) we get $f^{\text{PSC}}(\mathbf{x}^{\text{PSC}}) = Nx$ and a constraint of the form $x \geq \frac{1}{1 + (N-1)p} = \frac{1}{\mathbb{E}(C)}$ which implies that the optimal value is $\mathbf{x}^{\text{PSC}} = \frac{1}{\mathbb{E}(C)} \mathbf{1}_N$. This completes the proof.

APPENDIX E

PROOF OF THEOREM 6.1

Let $(\mathbf{x}^{\text{Opt}}, \mathbf{y}^{\text{Opt}})$ and \mathbf{x}^{PSC} be the optimal points for the LPs in (1) and (4), respectively. If \mathbf{x}^{Opt} is a feasible point in the LP in (4), then by definition

$$\bar{f}^{\text{PSC}} = f^{\text{PSC}}(\mathbf{x}^{\text{PSC}}) \leq f^{\text{PSC}}(\mathbf{x}^{\text{Opt}}) \leq f^{\text{Opt}}(\mathbf{x}^{\text{Opt}}, \mathbf{y}^{\text{Opt}}) = \bar{f}^{\text{Opt}},$$

where the second inequality holds by simply observing the objective functions of both problems.

Consider now the case where \mathbf{x}^{Opt} is not a feasible point in the LP in (4). Since it is feasible in (1), it satisfies

$$\mathbf{A}^{(k)} \mathbf{x}^{\text{Opt}} + \mathbf{y}_k^{\text{Opt}} \geq \mathbf{1}_N, \forall k \in [1 : K]. \quad (16)$$

Weighting (16) by p_k and taking the sum, we get

$$\sum_{k=1}^K p_k (\mathbf{A}^{(k)} \mathbf{x}^{\text{Opt}} + \mathbf{y}_k^{\text{Opt}}) = \mathbf{P} \mathbf{x}^{\text{Opt}} + \tilde{\mathbf{y}}^{\text{Opt}} \geq \mathbf{1}_N \quad (17)$$

where the equality holds by noticing that $\sum_{k=1}^K p_k \mathbf{A}^{(k)} = \mathbf{P}$

and by letting $\tilde{\mathbf{y}}^{\text{Opt}} = \sum_{k=1}^K p_k \mathbf{y}_k^{\text{Opt}} \geq \mathbf{0}_N$.

Consider now the point $\hat{\mathbf{x}}^{\text{PSC}} = \mathbf{x}^{\text{Opt}} + \tilde{\mathbf{y}}^{\text{Opt}}$. Then, this point is feasible in (4) since $\mathbf{P}\hat{\mathbf{x}}^{\text{PSC}} = \mathbf{P}\mathbf{x}^{\text{Opt}} + \mathbf{P}\tilde{\mathbf{y}}^{\text{Opt}} \geq \mathbf{P}\mathbf{x}^{\text{Opt}} + \tilde{\mathbf{y}}^{\text{Opt}} \geq \mathbf{1}_N$, where the first inequality holds because the diagonal entries of \mathbf{P} are all equal to 1, and all the other entries are non-negative, while the second inequality follows from (17). Thus, we get

$$\bar{f}^{\text{PSC}} \leq f^{\text{PSC}}(\hat{\mathbf{x}}^{\text{PSC}}) = f^{\text{Opt}}(\mathbf{x}^{\text{Opt}}, \mathbf{y}^{\text{Opt}}) = \bar{f}^{\text{Opt}}.$$

This completes the proof.

REFERENCES

- [1] M. A. Maddah-Ali and U. Niesen, "Fundamental limits of caching," *IEEE Trans. Inf. Theory*, vol. 60, no. 5, pp. 2856–2867, 2014.
- [2] J. Hachem, N. Karamchandani, and S. Diggavi, "Multi-level coded caching," in *2014 IEEE International Symposium on Information Theory (ISIT)*, pp. 56–60.
- [3] M. Ji, G. Caire, and A. F. Molisch, "Wireless device-to-device caching networks: Basic principles and system performance," *IEEE J. Sel. Areas Commun.*, vol. 34, no. 1, pp. 176–189, 2016.
- [4] G. Ananthanarayanan, V. N. Padmanabhan, L. Ravindranath, and C. A. Thekkath, "Combine: leveraging the power of wireless peers through collaborative downloading," in *Proceedings of the 5th international conference on Mobile systems, applications and services*. ACM, 2007, pp. 286–298.
- [5] L. Keller, A. Le, B. Cici, H. Seferoglu, C. Fragouli, and A. Markopoulou, "Microcast: cooperative video streaming on smartphones," in *Proceedings of the 10th international conference on Mobile systems, applications, and services*. ACM, 2012, pp. 57–70.
- [6] N. M. Do, C.-H. Hsu, J. P. Singh, and N. Venkatasubramanian, "Massive live video distribution using hybrid cellular and ad hoc networks," in *2011 IEEE International Symposium on a World of Wireless, Mobile and Multimedia Networks (WoWMoM)*, pp. 1–9.
- [7] B.-G. Chun, K. Chaudhuri, H. Wee, M. Barreno, C. H. Papadimitriou, and J. Kubiawicz, "Selfish caching in distributed systems: a game-theoretic analysis," in *Proceedings of the twenty-third annual ACM symposium on Principles of distributed computing*, 2004, pp. 21–30.
- [8] M. X. Goemans, L. Li, V. S. Mirrokni, and M. Thottan, "Market sharing games applied to content distribution in ad hoc networks," *IEEE J. Sel. Areas Commun.*, vol. 24, no. 5, pp. 1020–1033, 2006.
- [9] M. Taghizadeh, K. Micinski, C. Ofria, E. Torng, and S. Biswas, "Distributed cooperative caching in social wireless networks," *IEEE Trans. Mob. Comput.*, vol. 12, no. 6, pp. 1037–1053, 2013.
- [10] J. Reich and A. Chaintreau, "The age of impatience: optimal replication schemes for opportunistic networks," in *Proceedings of the 5th international conference on Emerging networking experiments and technologies*. ACM, 2009, pp. 85–96.
- [11] S. Ioannidis, L. Massoulie, and A. Chaintreau, "Distributed caching over heterogeneous mobile networks," in *ACM SIGMETRICS Performance Evaluation Review*, vol. 38, no. 1, 2010, pp. 311–322.
- [12] S. E. Tajbakhsh and P. Sadeghi, "Delay tolerant information dissemination via coded cooperative data exchange," *Journal of Communications and Networks*, vol. 17, no. 2, pp. 133–144, 2015.
- [13] F. Rebecchi, M. Dias de Amorim, V. Conan, A. Passarella, R. Bruno, and M. Conti, "Data offloading techniques in cellular networks: a survey," *IEEE Communications Surveys & Tutorials*, vol. 17, no. 2, pp. 580–603, 2015.
- [14] B. Han, P. Hui, V. A. Kumar, M. V. Marathe, J. Shao, and A. Srinivasan, "Mobile data offloading through opportunistic communications and social participation," *IEEE Trans. Mob. Comput.*, vol. 11, no. 5, pp. 821–834, 2012.
- [15] M. V. Barbera, A. C. Viana, M. D. De Amorim, and J. Stefa, "Data offloading in social mobile networks through vip delegation," *Ad Hoc Networks*, vol. 19, pp. 92–110, 2014.
- [16] S. Deb, M. Medard, and C. Choute, "Algebraic gossip: a network coding approach to optimal multiple rumor mongering," *IEEE Trans. Inf. Theory*, vol. 52, no. 6, pp. 2486–2507, 2006.
- [17] A. Pentland, N. Eagle, and D. Lazer, "Inferring social network structure using mobile phone data," *Proceedings of the National Academy of Sciences (PNAS)*, vol. 106, no. 36, pp. 15274–15278, 2009.
- [18] S. Kosta, A. Mei, and J. Stefa, "Small world in motion (swim): Modeling communities in ad-hoc mobile networking," in *2010 7th Annual IEEE Communications Society Conference on Sensor Mesh and Ad Hoc Communications and Networks (SECON)*, pp. 1–9.
- [19] P. Hui, A. Chaintreau, J. Scott, R. Gass, J. Crowcroft, and C. Diot, "Pocket switched networks and human mobility in conference environments," in *Proceedings of the 2005 ACM SIGCOMM workshop on Delay-tolerant networking*, pp. 244–251.
- [20] A. Chaintreau, P. Hui, J. Crowcroft, C. Diot, R. Gass, and J. Scott, "Pocket switched networks: Real-world mobility and its consequences for opportunistic forwarding," Technical Report UCAM-CL-TR-617, University of Cambridge, Computer Laboratory, Tech. Rep., 2005.
- [21] M. Karmoose, M. Cardone, and C. Fragouli, "Simplifying wireless social caching," in *2016 IEEE International Symposium on Information Theory (ISIT)*.
- [22] F. Ekman, A. Keränen, J. Karvo, and J. Ott, "Working day movement model," in *Proceedings of the 1st ACM SIGMOBILE workshop on Mobility models*, 2008, pp. 33–40.
- [23] D. Karamshuk, C. Boldrini, M. Conti, and A. Passarella, "Human mobility models for opportunistic networks," *IEEE Commun. Mag.*, vol. 49, no. 12, pp. 157–165, 2011.
- [24] M. C. Gonzalez, C. A. Hidalgo, and A.-L. Barabasi, "Understanding individual human mobility patterns," *Nature*, vol. 453, no. 7196, pp. 779–782, 2008.
- [25] S. Milgram, "The small world problem," *Psychology today*, vol. 2, no. 1, pp. 60–67, 1967.
- [26] P. Beraldi and A. Ruszczynski, "The probabilistic set-covering problem," *Operations Research*, vol. 50, no. 6, pp. 956–967, 2002.
- [27] A. Chaintreau, P. Hui, J. Crowcroft, C. Diot, R. Gass, and J. Scott, "Impact of human mobility on opportunistic forwarding algorithms," *IEEE Trans. Mob. Comput.*, vol. 6, no. 6, pp. 606–620, 2007.



Mohammed Karmoose is a Ph.D student in the Electrical Engineering department at UCLA. He received the BS and MS degrees in electrical engineering from the Faculty of Engineering in Alexandria University in Egypt in 2009 and 2013 respectively. He was a part of the CRN research group in E-JUST in Egypt as a Graduate Research Assistant from 2011 to 2014. He received the Annual Tribute Ceremony award for top-ranked students in Alexandria University in the years 2005 to 2009. He received the Electrical Engineering Department Fellowship from UCLA for his first year of Ph.D in 2014/2015. His research interests are distributed detection, cooperative caching and wireless communications.



Martina Cardone is currently a post-doctoral research fellow in the Electrical Engineering department at UCLA. She received the B.S. (Telecommunications Engineering) and the M.S. (Telecommunications Engineering) degrees summa cum laude from the Politecnico di Torino, Italy, in 2009 and 2011, respectively and the M.S. in Telecommunications Engineering from Télécom ParisTech, Paris, France, in 2011, as part of a double degree program. In 2015, she received the Ph.D. in Electronics and Communications from Télécom ParisTech (with work done at Eurecom in Sophia Antipolis, France). She received the second prize in the Outstanding Ph.D. award, Télécom ParisTech, Paris, France and the Qualcomm Innovation Fellowship in 2014. Her research interests are in network information theory, content-type coding, cooperative caching and wireless network secrecy.



Christina Fragouli is a Professor in the Electrical Engineering department at UCLA. She received the B.S. degree in Electrical Engineering from the National Technical University of Athens, Athens, Greece, in 1996, and the M.Sc. and Ph.D. degrees in Electrical Engineering from the University of California, Los Angeles, in 1998 and 2000, respectively. She has worked at the Information Sciences Center, AT&T Labs, Florham Park New Jersey, and the National University of Athens. She also visited Bell Laborato-

ries, Murray Hill, NJ, and DIMACS, Rutgers University. Between 2006–2015 she was faculty in the School of Computer and Communication Sciences, EPFL, Switzerland. She is an IEEE fellow, has served as an Information Theory Society Distinguished Lecturer, and as an Associate Editor for IEEE Communications Letters, Elsevier Journal on Computer Communication, IEEE Transactions on Communications, IEEE Transactions on Information Theory, and IEEE Transactions on Mobile Communications. Her research interests are in network coding, wireless communications, algorithms for networking and network security.